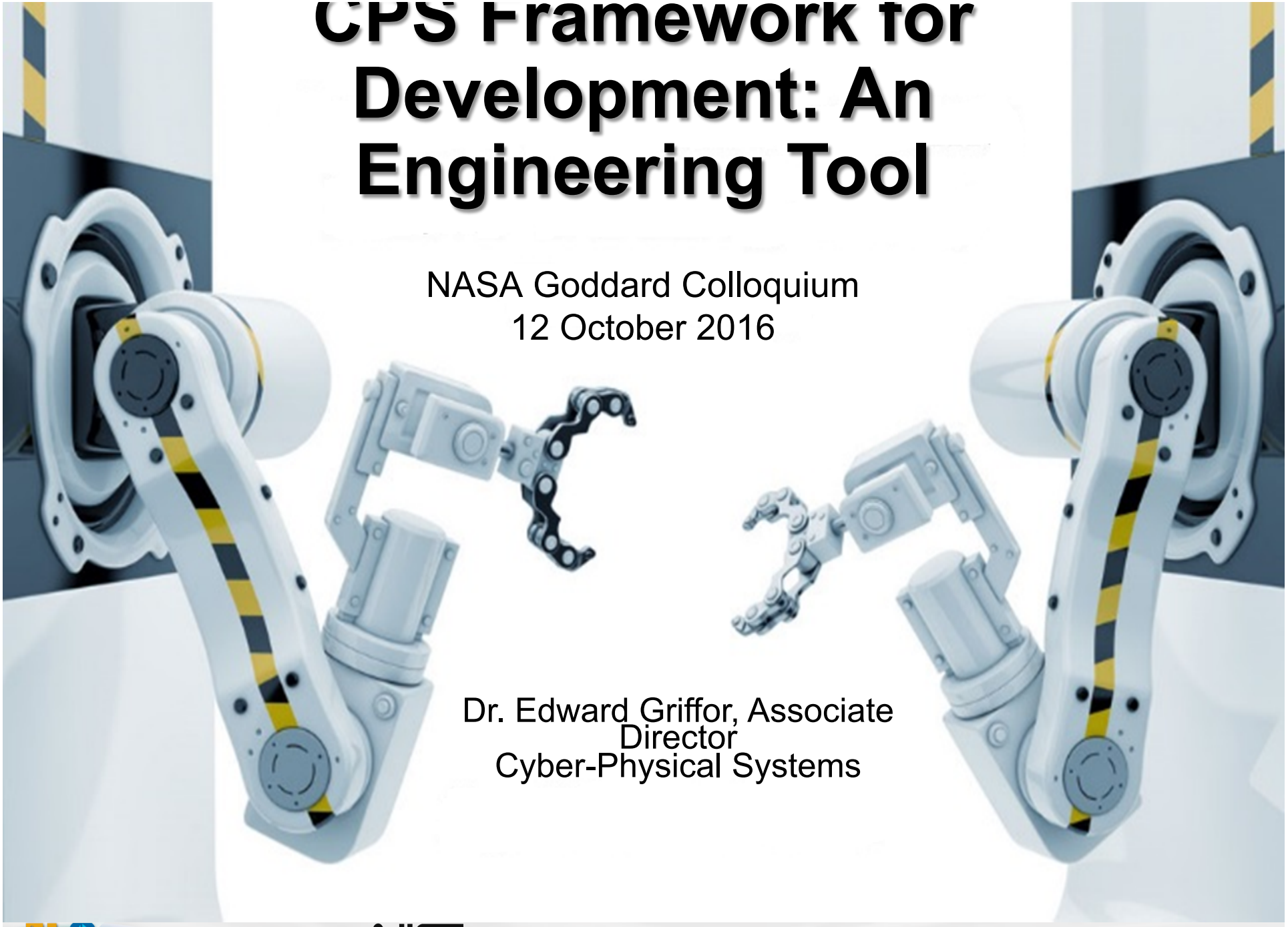
CPS Framework for Development: An Engineering Tool

NASA Goddard Colloquium
12 October 2016



Dr. Edward Griffor, Associate Director
Cyber-Physical Systems

Agenda

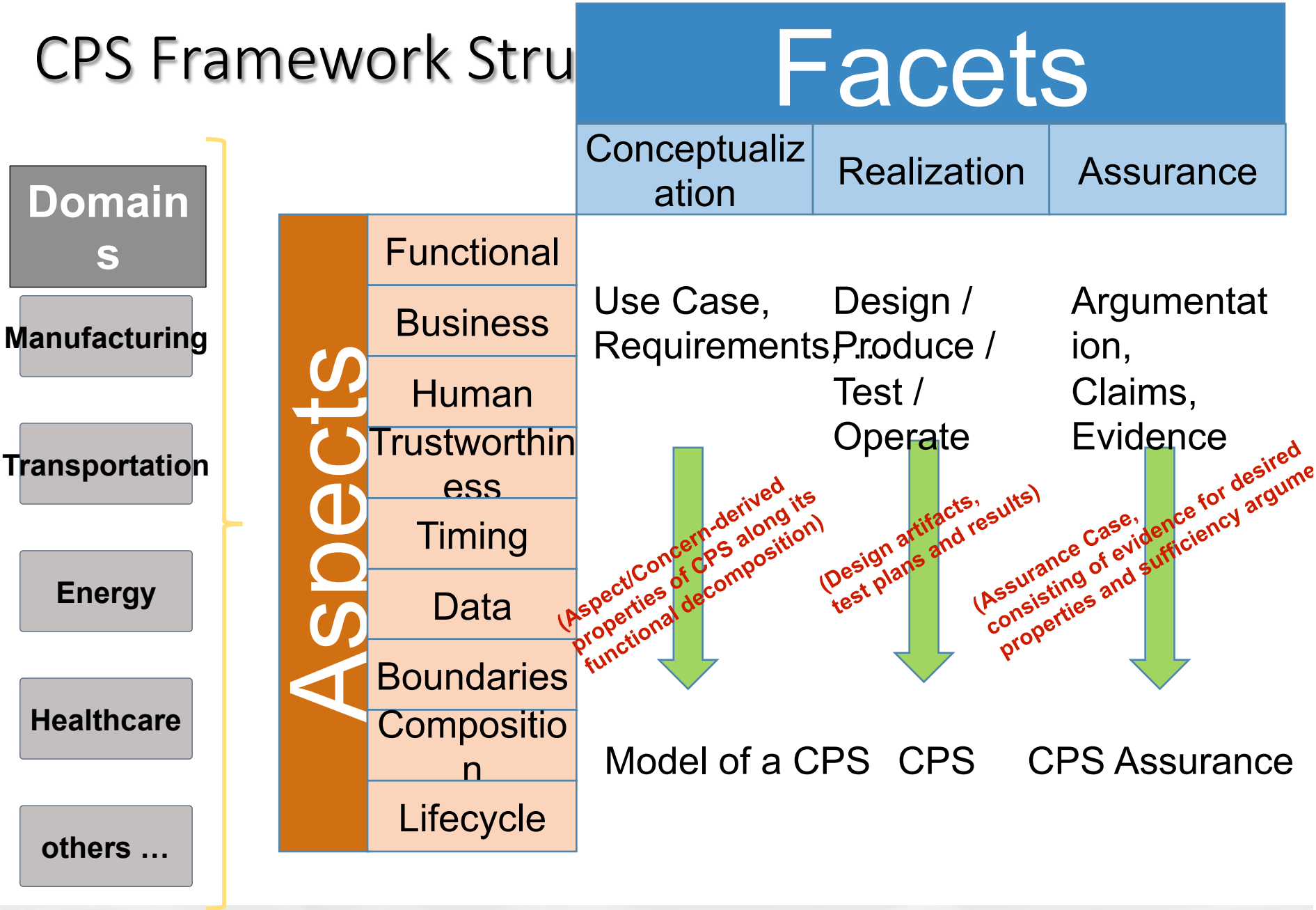
- CPS Framework: Purpose, Activities and Artifacts
- Overview of the CPS Framework Open Source Project
- Open Source Project: Models and Tools
- Back-Up and Notes

Purpose of the CPS Framework

- **Concern-driven structuring of development artifacts:** to facilitate assurance cases (by representing or analyzing a system along these dimensions, points of commonality or interoperability with other systems are revealed)
- **A normal-form for CPS/IoT system** (common way of presenting CPS/IoT that enables comparison of what is done, across the system, for the sake of any individual concern)
- Provides a **method for integrating CPS/IoT across domains** – the future of CPS/IoT is cross-domain integration. While some domains may have robust, integrated approaches to some concerns, there are typically radically different

CPS Framework is NOT A PROCESS!!
It is a method for integrating concerns into systems engineering

CPS Framework Stru



Framework Facets: Modes of Development

Facet	Description
Conceptualization	What things should be and what things are supposed to do: the set of activities that produce a model of a CPS (includes functional decomposition, requirements, and logical models.)
Realization	How things should be made and operate: the set of activities that produce, deploy, and operate a CPS (includes engineering tradeoffs and detailed designs in the critical path to the creation of a CPS instance.)
Assurance	How to achieve a desired level of confidence that things will work the way they should: the set of activities that provide confidence that a CPS performs as specified (includes claims, evidence, and argumentation.)

Conceptualization

Activity and Artifacts
Mission and Business Case Development **Artifact: Business use cases (feature content) with concern-driven properties like safety and security)**
Functional Decomposition **Artifact: Detailed use cases, actors/subsystems, influences (information and physical/energy exchanges)**
Requirements Analysis **Artifact: Functional and non-functional requirements (including**
Requirements Allocation **Artifact: Logical/Physical Configuration Items, HW/ SW Configuration Items, information/energy Configuration Items**
Interface Requirements Analysis **Artifact: Interface requirements**

Realization

Activity and Artifacts
Business Case Analysis **Artifact: Trade studies, lifecycle cost analysis, return on investment, and interdependencies with requirements, regulations, and incentives**
Lifecycle Management **Artifact: Lifecycle management and sustainability plan, integrated lifecycle management monitoring**
Design **Artifact: Design documentation, tradeoff analyses, requirement verification, virtual prototypes**
Manufacturing/Implementation **Artifact: Manufactured, integrated products, testing plans, and test results**
Operations **Artifact: Performance, quality, and product evolution tracking**
Disposal **Artifact: Reuse, sustainability and energy recovery assessments, disposal manifests**
Cyber-Physical Abstraction Layer Formation **Artifact: Domain (and product)-specific ontologies, modeling languages, and semantics specifications used in all phases of the lifecycle**
Phvsical Laver Realization

Assurance

Activity and Artifacts
Identify Assurance Objectives **Artifact: Assurance objectives/analysis report**
Define Assurance Strategy **Artifact: Strategy document/plan**
Control Assurance Evidence **Artifact: Control documentation**
Analyze Evidence **Artifact: Analysis report**
Provide Assurance Argument **Artifact: Assurance argument report**
Provide Estimate of Confidence **Artifact: Confidence estimate**
Configuration Audit **Artifacts: Product configuration assessment**
Requirements Verification **Artifact: Requirements and test results assessment**
Product Certification and Regulatory Compliance Testing **Artifact: Certifications**

Agenda

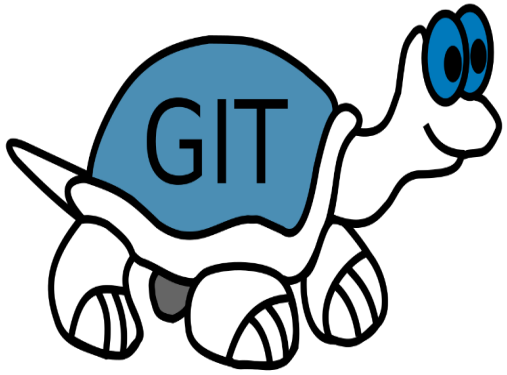
- CPS Framework: Purpose Activities and Artefacts
- Overview of the CPS Framework Open Source Project
- Open Source Project: Models and Tools
- Back-Up and Notes

Tools for Modeling the CPS Framework

Enterprise Architect: UML Editor

XMLSpy: XML/XMLSchema Editor

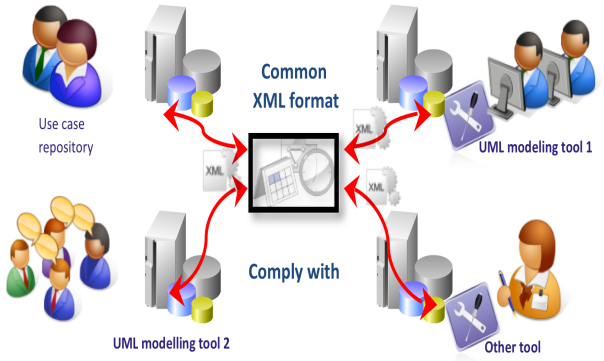
TortoiseGit: Windows GitTool



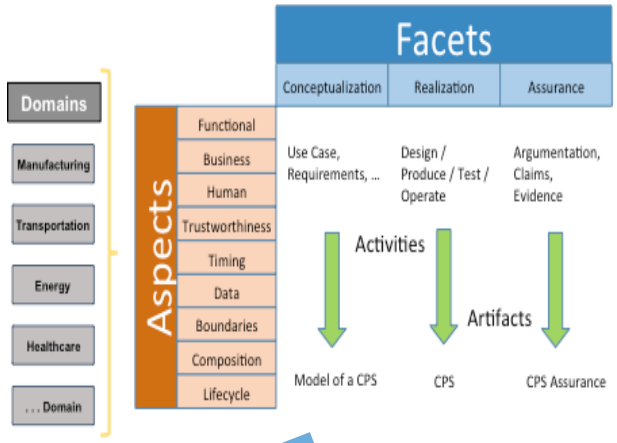
Notepadd++: Programmers Editor

Building a Model of a System in the Framework

IEC 62559 Methodology



NIST CPS Framework Methodology



Standardized XML Schema

Conceptualization	Realization	Assurance
•Business Case •Use Case •Requirements	•Design •Traceability to Requirements	•Algorithmically Prove Design Meets Requirements

MS Word-based Use Case Template

1 Description of the Use Case

1.1 Name of Use Case

Use case identification		
ID	*Area/ Domain(s)/ Zone(s)*	*Name of Use Case*

1.2 Version Management

Version management			
Version No.	*Date*	*Name Author(s)*	*Changes*

1.3 Scope and Objectives of Use Case

Scope and objectives of use case	
Scope	
Objective(s)	
Related business case(s)	

1.4 Narrative of Use Case

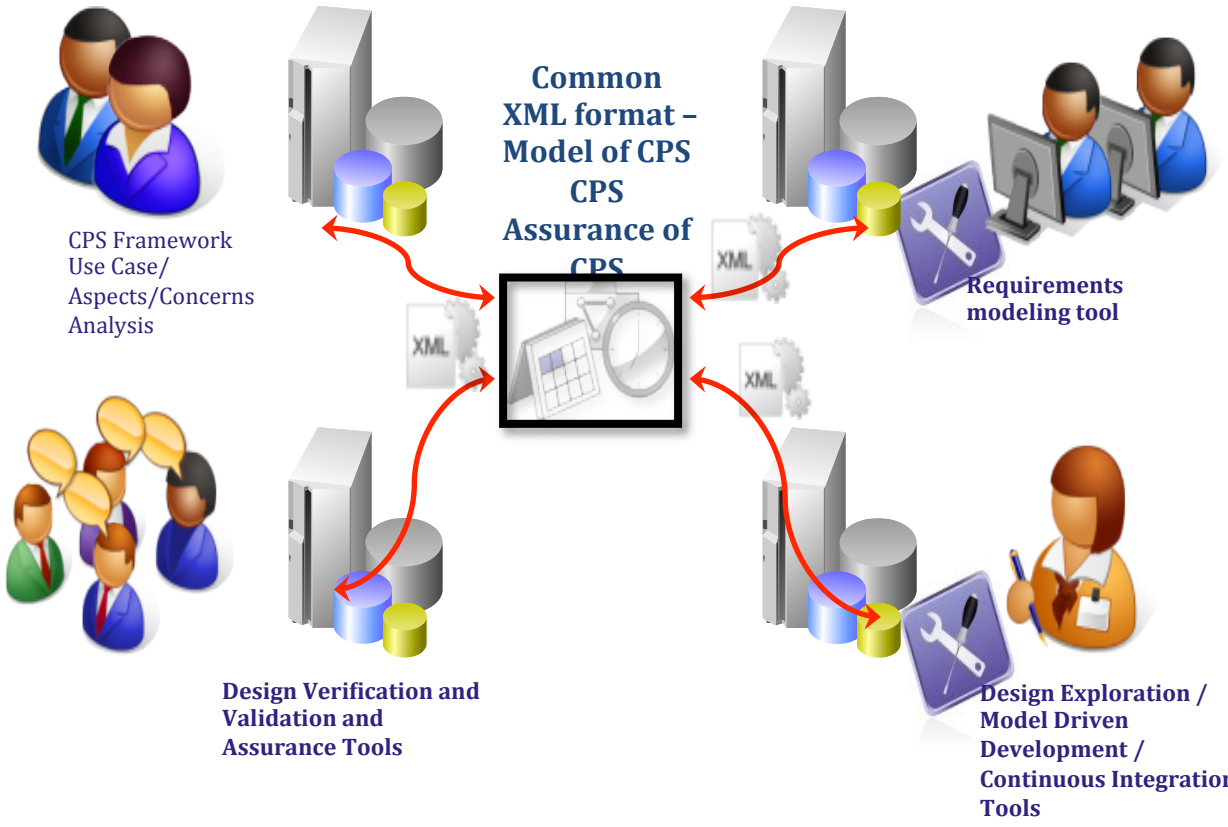
Narrative of use case
Short description
Complete description

Modeling Languages:
Auto-Import into Unified
Modeling Language (UML)

Agenda

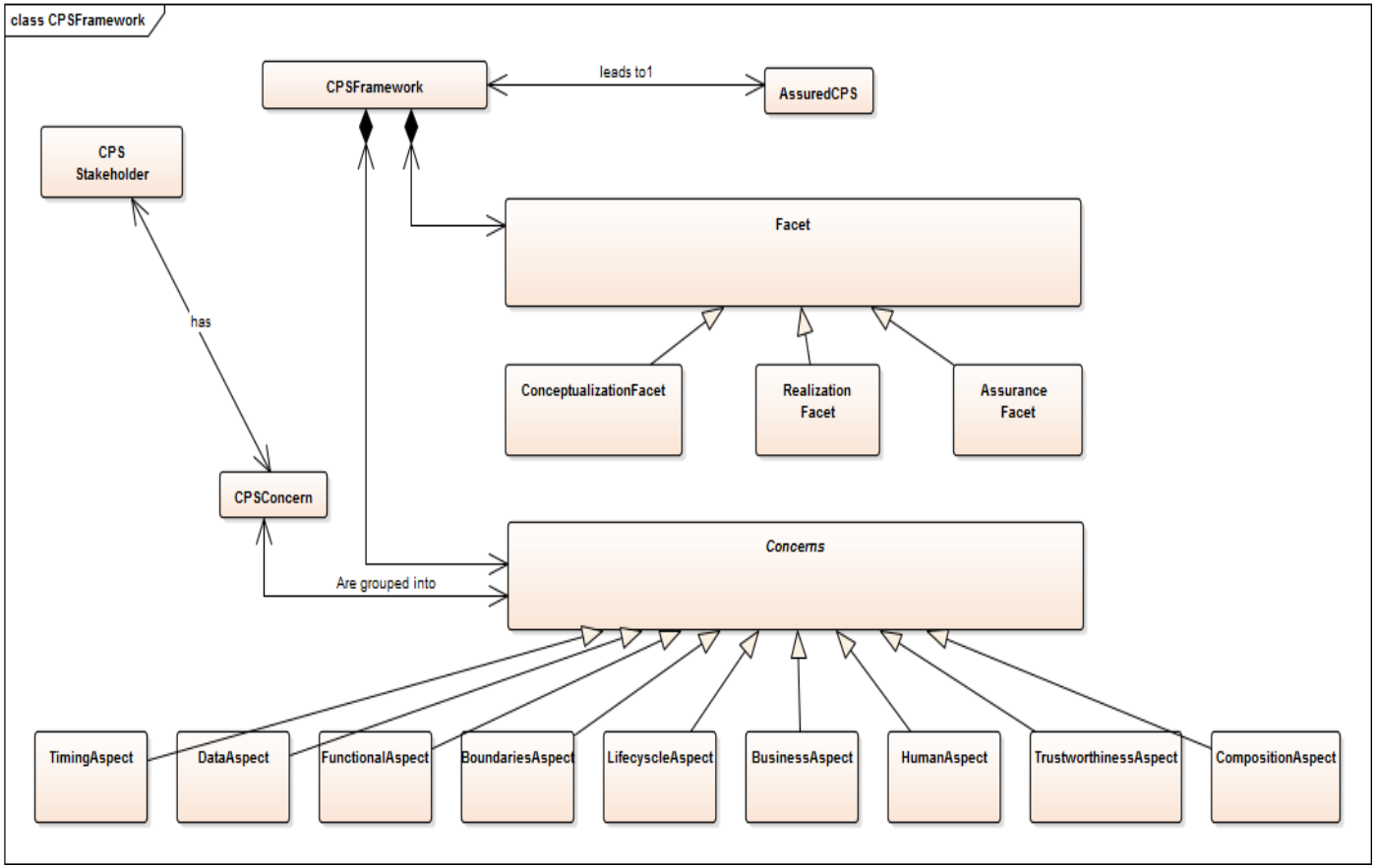
- CPS Framework: Purpose, Activities and Artefacts
- Overview of the CPS Framework Open Source Project
- Open Source Project: Models and Tools
- Back-Up and Notes

Engineering in the CPS Framework: One system representation, multiple views

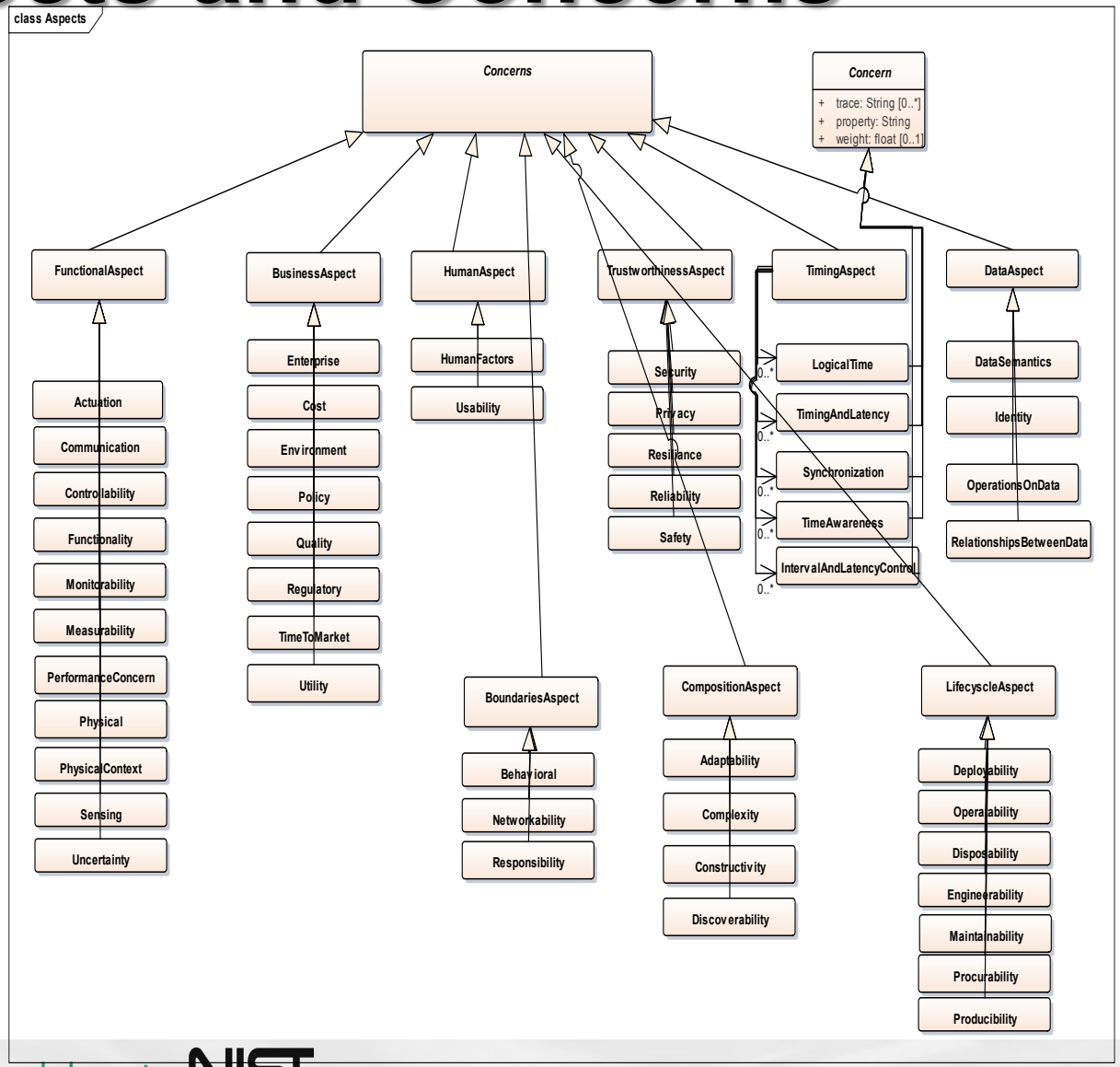


Markup Languages: Design and Engineering Data Exchange standard.

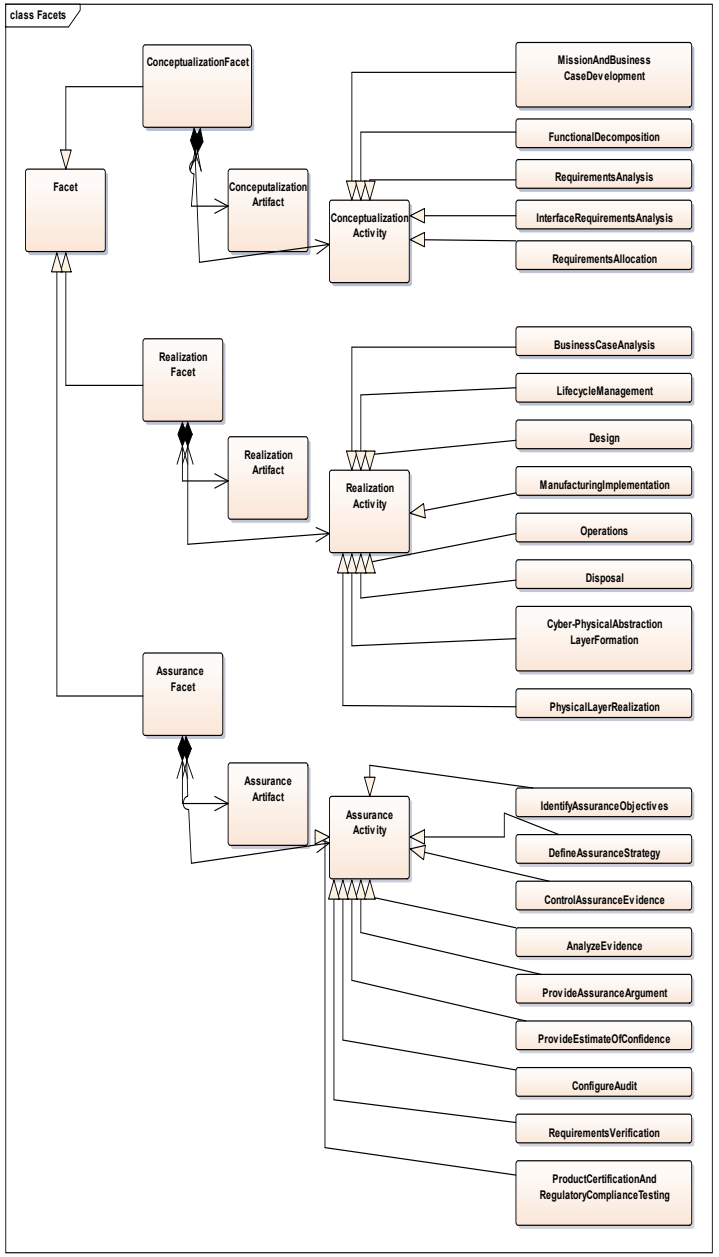
Modeling the CPS Framework: Aspects and Facets



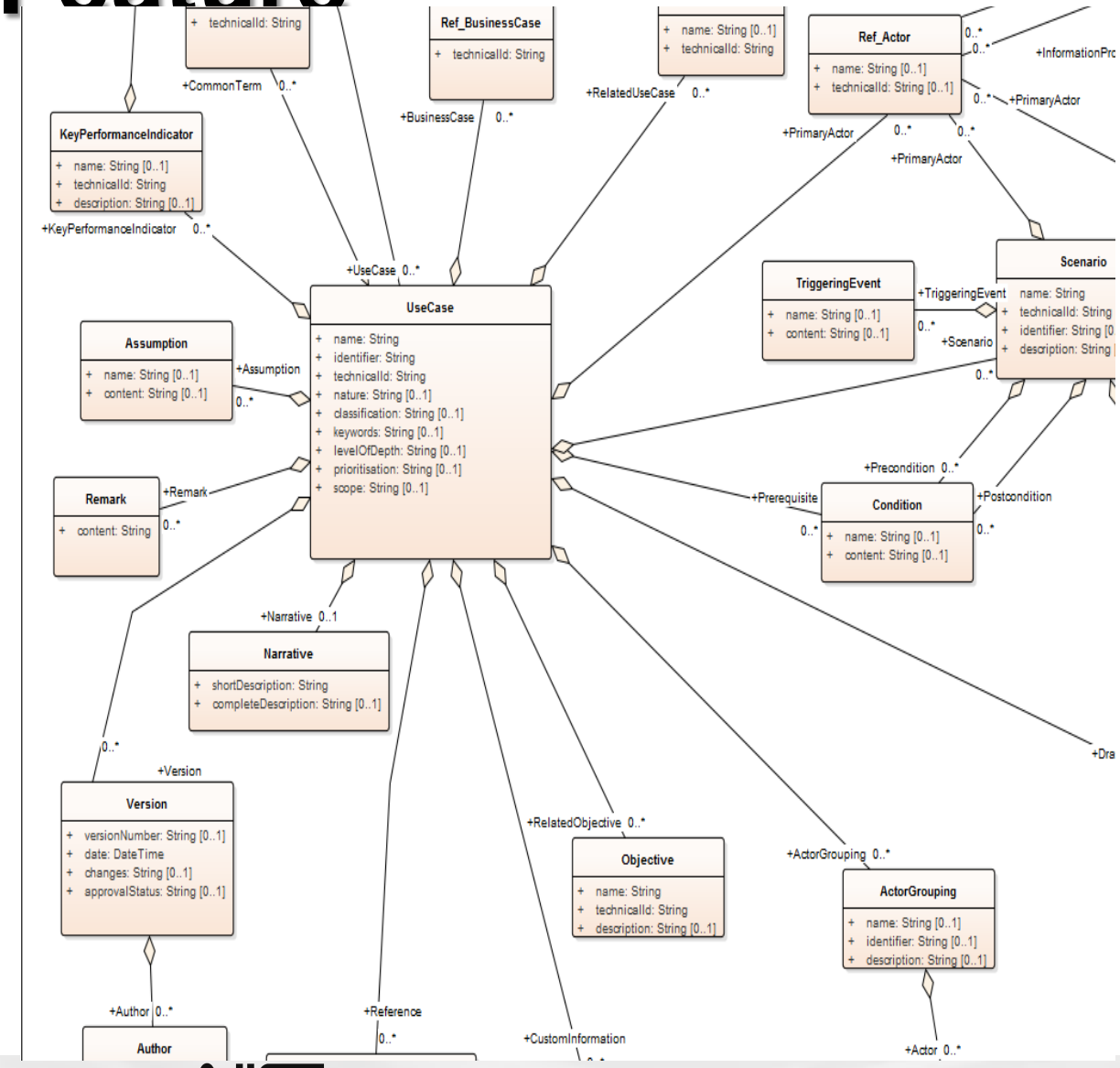
Modeling the Framework: Aspects and Concerns



Modeling the Framework: Facets and Activities

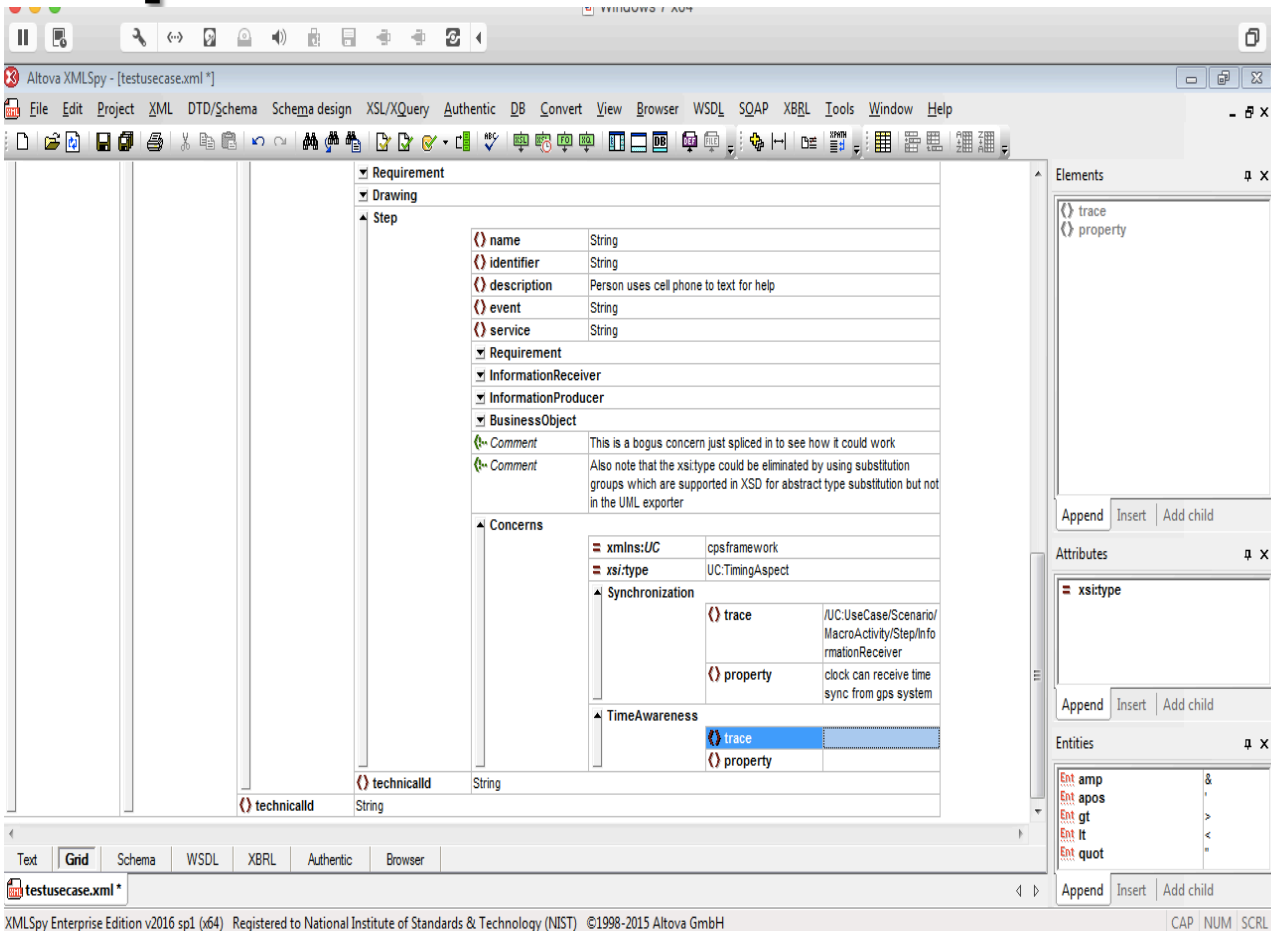


Modeling a Use Case*/System and Feature



*IEC 62559 Use Case Model

XML Editor: Concern-structured Requirements Development



Agenda

- Overview of the CPS Framework Open Source Project
- CPS Framework: Purpose, Activities and Artefacts
- Open Source Project: Models and Tools
- Back-Up and Notes

Takeaways:

- Important to note that the NIST CPS Framework and Open Source Implementation complements systems engineering (and other development processes). To apply the Framework requires tailoring to automotive: identifying relevant concerns and their metrics and integration methods/tools.
- The point of the CPS Framework is twofold:
  - the space of concerns related to the CPS/IoT domain and so context for any technology
  - relates the artifacts of development activities, providing a discipline for capturing the data required for the assurance case (system requirements, development artifacts and argumentation/reasoning)
- Industry 'recommended practices' for Functional